

Aplicativo Móvel para Ensino de Programação Utilizando Laboratório Remoto/

Mobile Application for Teaching Programming Using Remote Lab

DOI:10.34117/bjdv5n12-258

Recebimento dos originais: 10/11/2019

Aceitação para publicação: 18/12/2019

Lucas Dias Amaral

Graduado em Ciência da Computação pela Universidade Estadual do Sudoeste da Bahia

Instituição: Universidade Estadual do Sudoeste da Bahia

Endereço: Estrada do Bem Querer, Km 04 - 3293 - Bairro Universitário, Vitória da Conquista – BA, Brasil

E-mail: lucasamaralcc1@gmail.com

Maísa Soares dos Santos Lopes

Doutora em Engenharia Industrial pela Universidade Federal da Bahia

Instituição: Universidade Estadual do Sudoeste da Bahia

Endereço: Estrada do Bem Querer, Km 04 - 3293 - Bairro Universitário, Vitória da Conquista – BA, Brasil

E-mail: maisa@uesb.edu.br

RESUMO

A complexidade que envolve o ensino e aprendizado de programação em disciplinas introdutórias estabelece uma barreira para professores e alunos com relação a transmissão e absorção de conteúdo. A experimentação desponta como uma ferramenta pedagógica de ensino, capaz de encurtar a distância entre essas disciplinas e a realidade do aluno, por conta de sua característica visual. Este artigo tem como objetivo descrever o desenvolvimento de um aplicativo que dá suporte ao ensino de programação, por meio do qual seja possível manipular um robô móvel remoto através de algoritmos e visualizar a execução dos códigos em tempo real. O aplicativo foi desenvolvido com uso de HTML, CSS3, JavaScript e do framework Apache Cordova. O método de avaliação heurística de usabilidade foi utilizado na avaliação da aplicação, que apresentou alta usabilidade.

Palavras chaves: Laboratório Remoto; Aprendizagem Móvel; Desenvolvimento Híbrido; Usabilidade Mobile.

ABSTRACT

The complexity involved in teaching and learning programming in introductory subjects establishes a barrier for teachers and students regarding the transmission and absorption of content. The experimentation emerges as a pedagogical tool of teaching, able to shorten the distance between these disciplines and the reality of the student, due to its visual characteristic. This article aims to describe the development of an application that supports the teaching of programming, through which it is possible to manipulate a remote mobile robot through algorithms and visualize the execution of the codes in real time. The application was developed using HTML, CSS3, JavaScript and the Apache Cordova framework. The heuristic usability evaluation method was used in the evaluation of the application, which presented high usability.

Keywords: Remote laboratory; Mobile learning; Hybrid development; Mobile Usability.

1 INTRODUÇÃO

A experimentação é de grande importância para despertar o interesse de alunos pela ciência e facilitar a compreensão de seus fenômenos, uma vez que, sua utilização encurta a distância entre a teoria e a realidade vivida por cada um, favorecendo desse modo a absorção dos conceitos científicos. Ainda assim, são poucas as escolas e universidades que podem contar com laboratórios convencionais e infraestrutura para comportar experimentos com qualidade, em virtude do alto custo para construção e manutenção dos mesmos.

Dados do último Censo Escolar do Inep (2018) apontam que, enquanto o acesso à internet é uma realidade em 95,1% das escolas de ensino médio, o laboratório de ciências, é encontrado em apenas 44,1% delas. Com relação às universidades, em 2018, a verba relativa a investimento, aquisição de bens, equipamentos e obras de expansão e reestruturação das instituições sofreu cortes, chegando a, aproximadamente, 10% daquela destinada às instituições em 2014 [Santos & Soares 2017]. Como consequência da falta de recursos, as universidades se veem sujeitas a fechar seus laboratórios, que por vezes, se encontram sucateados por conta da falta de recursos.

A Experimentação Online desponta como uma alternativa de baixo custo perante a escassez de laboratórios convencionais nas escolas e universidades. Seu conceito compreende os laboratórios virtuais e remotos, favorecido por ferramentas online baseadas em realidade virtual, realidade aumentada, dispositivos sensoriais, vídeos ao vivo, vídeos interativos e jogos que promovem a imersão do usuário em ambientes virtuais [Restivo & Cardoso 2013]. Tal concepção visa recriar experiências reais, seja por meio de simulações

ou acesso remoto, de modo que, o aluno possa visualizar a aplicação da teoria e usufruir dos benefícios da experimentação.

O desenvolvimento da Experimentação Online é muito significativo, visto que, matérias que costumam ser obstáculos para professores e alunos, como é o caso das disciplinas introdutórias que envolvem algoritmos e programação em cursos superiores e técnicos, devido a sua complexidade, demandam por suporte experimental, uma vez que, exigem o desenvolvimento de estratégias de solução de problemas com base lógico-matemática, que para muitos são altamente abstratas e distantes do “cotidiano” [Raabe & Silva 2005]. Os sistemas visuais, como é o caso da experimentação e dos simuladores, apresentam um grande potencial como ferramenta de ensino da programação, pois, algoritmos computacionais podem ser pouco claros quando apresentados em formato textual [Gomes 2010].

Diante desse cenário, os avanços tecnológicos vêm provocando profundas mudanças no âmbito dos processos de ensino e aprendizagem. Uma das principais razões disso é a ascensão dos dispositivos móveis, que independente de algumas limitações, apresentam vantagens como: acessibilidade, conectividade, comodidade e portabilidade. A integração entre esses benefícios e concepções educacionais como a Experimentação Online, estabelece novos parâmetros com relação à inclusão digital no ensino e aprendizagem.

O presente trabalho tem como objetivo descrever o desenvolvimento de um aplicativo interativo e fácil de usar para dispositivos móveis, que permite manipular o robô móvel remoto do projeto LARA (Laboratório Remoto em AVA) [Lopes et al. 2016], o LaraApp. Por meio da ferramenta, o usuário pode criar um programa utilizando estruturas sequenciais, de seleção e de repetição que será executado pelo robô. O aplicativo visa também atrair os alunos para ensino/aprendizagem de programação de computadores tornando este processo mais dinâmico.

2 APRENDIZAGEM MÓVEL

A Aprendizagem móvel (M-learning) é a aprendizagem mediante o uso de dispositivos móveis do tipo telefones móveis básicos, tablets, PDAs, reprodutores MP3, memórias USB, dispositivos de leitura eletrônica e Smartphones [Silva et al. 2013]. A ideia se baseia em prover conteúdos educacionais, disponibilizando-os por intermédio de dispositivos móveis.

Este tipo de aplicação concede ao aluno a oportunidade de realizar atividades educacionais a qualquer hora e local, assegurando, de tal forma, maior flexibilidade no acesso aos conteúdos, simulações ou experimentos reais manipulados à distância.

Por intermédio da Aprendizagem Móvel, é possível aproximar métodos de ensino e aprendizagem da realidade do aluno já que este vivencia uma era tecnológica, no entanto, os dispositivos móveis por si só não se constituem em ferramentas educativas úteis, tornando-se imprescindível a pesquisa sobre o uso pedagógico destes [Silva et al. 2013].

3 EXPERIMENTAÇÃO REMOTA

A experimentação remota permite ao usuário alterar parâmetros de controle, executar experiências, observar resultados em tempo real (capturados por câmeras de vídeo) e baixar dados através de uma interface web [Casini et al. 2003]. Por meio desse conceito, instituições de ensino que possuem laboratórios convencionais, podem disponibilizar seus experimentos remotamente, permitindo desta forma que qualquer estudante com acesso à internet possa manipular os mesmos.

Abaixo são listadas algumas vantagens que os Laboratórios Remotos podem oferecer aos seus usuários em comparação aos laboratórios virtuais e convencionais [Nedic et al. 2003].

- Acesso remoto ao laboratório;
- Realizar experimentos em equipamentos reais;
- Colaborar;
- Aprender por tentativa e erro;
- Realizar análises em dados experimentais reais;
- Flexibilidade na escolha de horário e local para a realização de experimentos.

Os Laboratórios Remotos levam o estudante a vivenciar uma experiência direta com o processo real [Casini et al. 2003], fator importante para o aprendizado das ciências e engenharias, sendo uma de suas desvantagens, o fato de proporcionar não mais que uma presença virtual no laboratório [Nedic et al. 2003].

3.1 EXPERIMENTAÇÃO REMOTA MÓVEL

A Experimentação Remota Móvel (Mobile Remote Experimentation, MRE), herda os conceitos e características dos Laboratórios Remotos, sendo assim, destinam-se a

disponibilizar acesso a experimentos reais por meio de dispositivos móveis, como smartphones e tablets. Experimentos reais acessados remotamente por intermédio de dispositivos móveis, proporcionam aos estudantes uma nova maneira de interagir com as disciplinas de forma simples e agradável, em qualquer lugar e a qualquer momento [Silva et al. 2013].

A popularidade dos smartphones, tende a ampliar a perspectiva de acesso aos Laboratórios Remotos. Atualmente, estes aparelhos com alto poder de processamento, leves e fáceis de transportar e munidos de diversas alternativas de acesso a internet (3g, 4g, Wi-Fi, entre outras.), são realidade para a maioria dos estudantes por conta dos preços acessíveis. Suas principais vantagens são a disponibilidade de acesso e a notável portabilidade, essas características, aliadas às tecnologias de redes sem fio (wireless), permitem ao estudante realizar experimentos sem qualquer restrição temporal e/ou geográfica (casa, universidade, hotéis, transporte público, etc.) [Costa 2005].

Apesar das vantagens mencionadas, algumas características inerentes aos dispositivos móveis, podem limitar o desenvolvimento de aplicações para MRE. Dentre elas estão: tamanho reduzido da tela, internet por vezes limitada, ausência de teclado amigável, bateria com pouca autonomia, memória reduzida, entre outras [Costa 2005]. Estas características estabelecem os dispositivos móveis, preferencialmente, como auxiliares ou alternativa de expansão para os Laboratórios Remotos, visto que, os computadores pessoais, ainda dispõem de qualidades significativas no que diz respeito a experimentação a distância.

4 APRESENTAÇÃO DO LARAAPP

Para desenvolvimento do LaraApp, foi utilizado um processo iterativo incremental. O levantamento de requisitos se baseou na análise das funcionalidades do LARA, da API do robô e em entrevistas com alunos e professores. A modelagem foi realizada utilizando a Linguagem de Modelagem Unificada (UML). Para a implementação do aplicativo, foram utilizadas tecnologias padrão da web (HTML, CSS3 e JavaScript) e a framework de código aberto Apache Cordova, capaz de compilar aplicações HTML/JavaScript para diversas plataformas, incluindo Android e iOS.

A Figura 1 apresenta o diagrama de casos de uso do LaraApp, onde o usuário interage com a aplicação, mediante possíveis ações indicadas nas elipses.

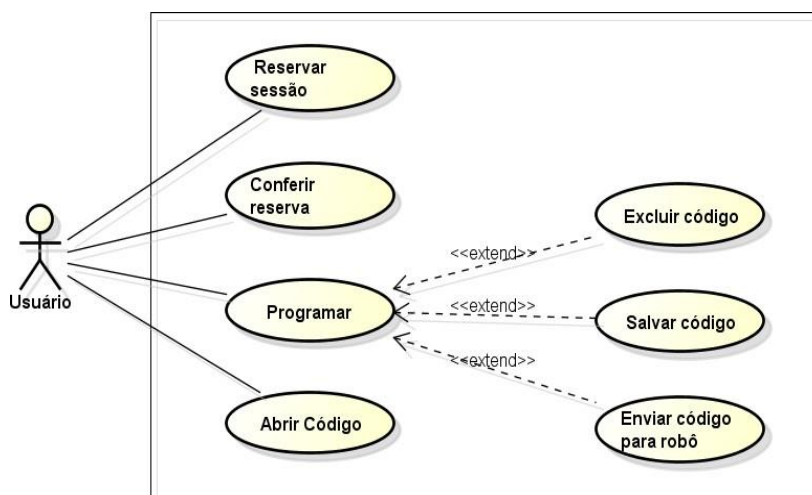
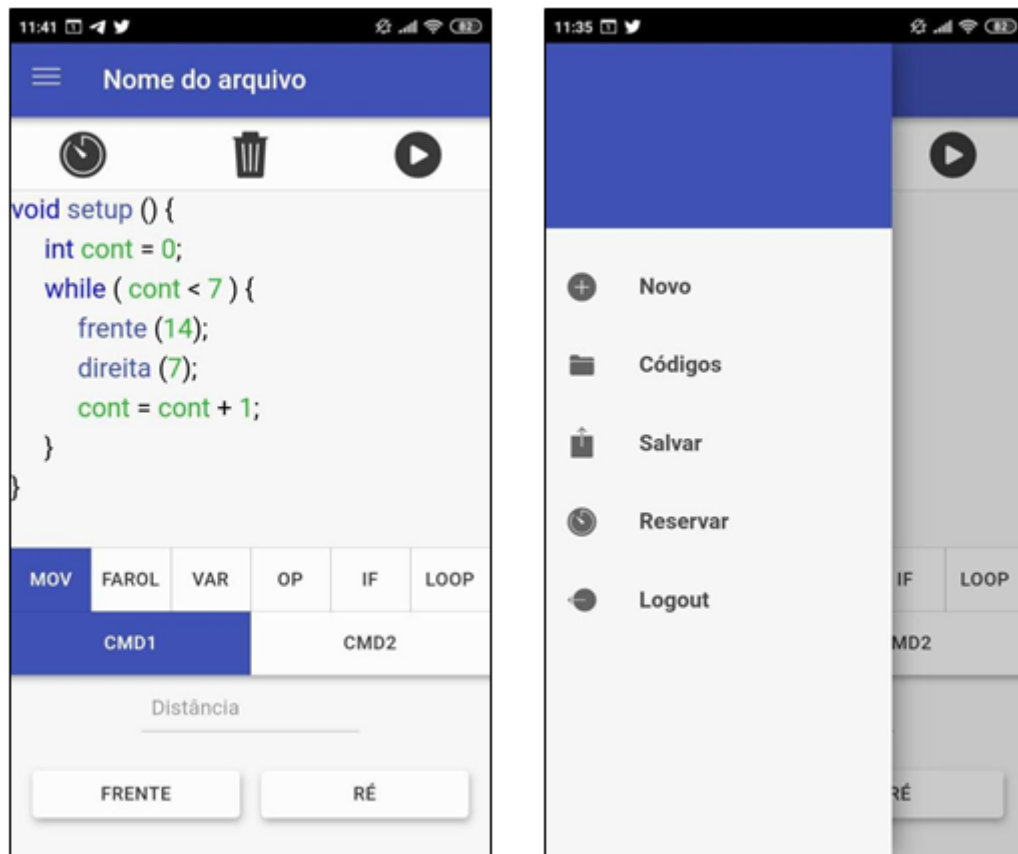


Figura 1. Diagrama de casos de uso

Ao abrir o aplicativo, o usuário tem acesso a tela de programação, na qual pode-se combinar as estruturas e comandos de programação para gerar códigos, salvar algoritmos e abrir códigos salvos anteriormente. Caso o usuário tenha interesse em visualizar a execução dos seus códigos pelo robô do LARA, uma reserva deve ser feita pelo mesmo, com isso, durante o tempo de reserva, o usuário poderá enviar códigos e observar o comportamento do robô em tempo real. Além disso, o usuário pode verificar quanto tempo de reserva lhe resta

4.1 INTERFACE DO LARAAPP

Durante a fase de modelagem, alguns protótipos foram criados, com o intuito de obter um modelo de interface que melhor se adequasse ao objetivo do trabalho. Em virtude de ser uma ferramenta de apoio a iniciantes em programação, o aplicativo foi planejado com o propósito de possuir uma interface simples e intuitiva. Desta forma, evita-se que o aluno se desprenda do objetivo principal da ferramenta, enquanto resolve problemas relativos ao manuseio da interface. A Figura 2 exibe a tela de programação (Figura 2a) e a barra lateral do aplicativo (Figura 2b).



(a)

(b)

Figura 2. Screenshots do LaraApp

Na tela de programação, o usuário pode navegar por abas que contém comandos de movimento do robô (frente, ré, direita, esquerda, ligar e desligar farol) e estruturas básicas de programação (declarar variável, realizar operações matemáticas, comando de seleção e comando de repetição). O usuário deve combinar os comandos e estruturas para gerar seu código, o qual é exibido na área de texto, como é visto na Figura 2a. A barra superior com ícones, permite ao usuário conferir sua reserva, apagar o código contido na área de texto e enviar algoritmo para o robô.

Na barra lateral do aplicativo, exibida na Figura 2b, o usuário pode criar um novo código, abrir códigos salvos, salvar códigos e reservar uma sessão para ter acesso ao robô do LARA.

A fim de exigir o mínimo possível de entradas a partir do teclado, o sistema com abas e botões foi planejado com o propósito de evitar uma experiência exaustiva, por conta do teclado pouco amigável característico dos smartphones. Apesar dessa limitação, o aplicativo

permite inúmeras combinações, que podem ser exploradas pelo usuário durante a criação de algoritmos.

5 AVALIAÇÃO DO LARAAPP

Para avaliar o LaraApp, optou-se pela aplicação da avaliação heurística de usabilidade, por ser um teste eficaz e exigir poucos recursos. O processo contou com três avaliadores especialistas, cada um deles com conhecimento tanto no que diz respeito ao domínio da aplicação, quanto para análise de interfaces com base em padrões de usabilidade predefinidos.

O conjunto de heurísticas de usabilidade tomado como base para essa avaliação foi proposto pelo Grupo de Qualidade do Software da Universidade Federal de Santa Catarina (GQS/INSCoD/UFSC). Desenvolvido a partir das dez heurísticas de Nielsen (1994), o instrumento disponibiliza um checklist (lista de verificação) com 48 questões, gerado por meio da adaptação dos princípios de usabilidade tradicionais, para as interfaces de celulares touchscreen. O checklist funciona como medidor da qualidade da interface sendo avaliada. Após o preenchimento do questionário por parte do avaliador, o resultado é exibido, no qual a classificação de qualidade da interface é revelada, juntamente com uma breve descrição, apontando os princípios de usabilidade que foram violados, ou não, pela aplicação. O checklist para avaliação da usabilidade de aplicativos para celulares touchscreen (MATCh) pode ser acessada em: <http://match.inf.ufsc.br:90>.

Ao fim da avaliação, obteve-se os resultados exibidos na Tabela 1.

Tabela 1. Resultados da avaliação do LaraApp pelo checklist

Avaliador	Resultado
Avaliador 1	58.6 pontos - Usabilidade alta
Avaliador 2	60.9 pontos - Usabilidade muito alta
Avaliador 3	61.7 pontos - Usabilidade muito alta

Além dos resultados retornados pelo checklist, obteve-se as recomendações dos avaliadores sobre possíveis soluções para problemas de usabilidade na interface do LaraApp.

A tabela 2 expõe as falhas identificadas no decorrer da inspeção, a gravidade de cada problema e as soluções propostas pelos avaliadores.

A gravidade é definida a partir da prioridade de cada problema: 1 - Prioridade muito baixa: caso não haja tempo de sobra para correção, esse tipo de problema pode ser ignorado a princípio; 2 - Prioridade baixa: é desejável que seja consertado, porém recebe baixa prioridade; 3 - prioridade alta: é importante que seja corrigido, se possível, antes do lançamento; e 4 - Prioridade muito alta: é imprescindível que seja solucionado antes do lançamento do sistema.

Verifica-se na tabela 2, que os avaliadores identificaram cinco problemas de usabilidade no LaraApp, sendo que dois apresentam gravidade 1, um apresenta gravidade 2, e por fim, dois apresentam gravidade 3.

As falhas de usabilidade de gravidade 1 foram apontados como sendo as labels “FAROL ON” e “FAROL OFF” dos botões da aba “FAROL” e o ícone pouco comum para a ação “salvar”, localizado na barra lateral da tela de programação. Segundo os avaliadores, a combinação de idiomas das labels na aba “FAROL”, podem causar confusão, sendo assim, foi indicado que as labels fossem alteradas para “LIGAR” e “DESLIGAR”, evitando desse modo, a combinação de dois idiomas distintos. Já com relação ao ícone “salvar”, os avaliadores sugeriram que fosse trocado por um ícone mais comum, uma vez que o ícone utilizado não é tão intuitivo.

Tabela 2. Problemas, correções propostas e prioridade para correção

Problema	Gravidade	Proposta de correção
As labels dos botões “FAROL ON” e “FAROL OFF” da aba farol com nomes que combinam dois idiomas.	1	Alterar os nomes dos botões da aba farol para “LIGAR” e “DESLIGAR”.
Ícone que indica “salvar” pouco intuitivo.	1	Utilizar um ícone mais comum para “salvar”.
Complexidade para utilizar comandos mais robustos: VAR, OP, IF e WHILE.	2	Adicionar um ícone de ajuda que abre um pop-up com dicas de como usar os comandos.
Não indica em que linha no código o cursor do usuário está em cada momento.	3	Destacar a linha de código em que o cursor <u>está aplicando</u> uma cor diferente nesta posição.
Dificuldade em esconder o teclado após ter digitado em um input (entrada).	3	Fazer com que o botão avançar do teclado, quando acionado, esconda o mesmo.

A falha de usabilidade de gravidade 2, está relacionada a complexidade que envolve o uso das estruturas de programação (VAR, OP, IF e WHILE) através da aplicação, para os avaliadores, um sistema de ajuda deve ser implementado, de modo que o usuário iniciante possa observar exemplos de como utilizar os comandos mais complexos.

Por fim, as falhas de usabilidade de gravidade 3, segundo os avaliadores, foram a falta de feedback com relação a linha em que se encontra o cursor do usuário em seu código após realizar alguma ação e a dificuldade em esconder o teclado, após o uso do mesmo. Como resolução para o primeiro problema de gravidade 3, os avaliadores propuseram que a linha onde se encontra o usuário, tenha uma coloração distinta das demais, de forma a destacar essa posição, impedindo de tal maneira, que o usuário se perca enquanto constrói algoritmos mais complexos. Já o segundo problema de gravidade 3, deve ser sanado, adicionando ao botão “avançar” do teclado, a função de esconder o próprio, de modo a impedir possíveis incômodos no decorrer do processo de programação.

6 CONCLUSÃO

O presente trabalho apresenta uma aplicação desenvolvida com a finalidade de facilitar o ensino e aprendizagem de programação através do conceito de experimentação remota. Essa ferramenta é composta por um aplicativo para dispositivos móveis, utilizado para criação de algoritmos de controle e um robô remoto móvel encarregado de executar o código.

A interface de programação do LaraApp, baseada em abas e botões, se mostrou bastante eficaz para prevenir que o usuário digite grandes quantidades de texto enquanto programa. Além disso, com relação ao suporte no ensino e aprendizagem de disciplinas que envolvem programação, o aplicativo mostrou ter potencial para contribuir como ferramenta de apoio aos iniciantes, além de figurar como alternativa perante a escassez de laboratórios convencionais nas escolas e universidades. Porém, até então, o desenvolvimento da aplicação se encontra em andamento, uma vez que, a conexão com o robô do Lara e o sistema de controle de reservas, estão em fase de finalização.

A respeito de trabalhos futuros, além do aprimoramento da interface, com base no feedback dos usuários e da avaliação do LaraApp como ferramenta de ensino e aprendizagem de programação, também existe a possibilidade de expansão da aplicação, mediante a

integração com um laboratório virtual capaz de trabalhar com a linguagem de programação compatível com o robô do Lara.

REFERÊNCIAS

- Casini, M., Prattichizzo, D., & Vicino, A. (2003). E-learning by remote laboratories: A new tool for control education. *IFAC Proceedings Volumes*, 36(10), 73-78.
- Costa, R. (2005). Tele-Experimentação Móvel (Mobile Remote Experimentation)- Considerações sobre uma área emergente no ensino à distância. *Journal of scientific activity at ISEP*, 15.
- Gomes, A. D. J. (2010). *Dificuldades de aprendizagem de programação de computadores: contributos para a sua compreensão e resolução* (Doctoral dissertation).
- Inep (2019). Dados do Censo Escolar – Noventa e cinco por cento das escolas de ensino médio têm acesso à internet, mas apenas 44% têm laboratório de ciências. Disponível em: <http://portal.inep.gov.br/artigo/-/asset_publisher/B4AQV9zFY7Bv/content/dados-do-censo-escolar-noventa-e-cinco-por-cento-das-escolas-de-ensino-medio-tem-acesso-a-internet-mas-apenas-44-tem-laboratorio-de-ciencias/21206/>. Acesso em: 5 de jun. de 2019.
- Lopes, M., Gomes, I., Trindade, R., Silva, A., & Lima, A. C. (2016). Web environment for programming and control of mobile robot in a remote laboratory. *IEEE Transactions on Learning Technologies*, 10(4), 526–531.
- Nedic, Z., Machotka, J., & Nafalski, A. (2003). *Remote laboratories versus virtual and real laboratories* (Vol. 1, pp. T3E-T3E). IEEE.
- Nielsen, J. (1994). Usability inspection methods. In *Conference companion on Human factors in computing systems* (pp. 413-414). ACM.
- Raabe, A. L. A., & Silva, J. D. (2005). Um ambiente para atendimento as dificuldades de aprendizagem de algoritmos. In *XIII Workshop de Educação em Computação (WEI'2005)*. São Leopoldo, RS, Brasil.
- Restivo, M. T., & Cardoso, A. (2013). Exploring online experimentation. *International Journal of Online Engineering (iJOE)*, 9(S8), 4-6.
- Santos, M., & Soares, I. (2018). Com universidades sucateadas, Brasil cai nos ranking globais de avaliação. Disponível em:

<<https://www.correiobraziliense.com.br/app/noticia/brasil/2017/12/25/interna-brasil,649776/com-universidades-sucateadas-brasil-cai-nos-ranking-globais-de-avalia.shtml/>>. Acesso em: 6 de jun. de 2019.

Silva, J. B., Rochadel, W., Simão, J. P., & da Silva Fidalgo, A. V. (2013). Uso de dispositivos móveis para acesso a Experimentos Remotos na Educação Básica. *VAEP-RITA*, 1(2), 129-134.